Información sobre Datos Agregados mediante las Funciones de Grupo

Objetivos

Al finalizar esta lección, debería estar capacitado para:

- Identificar las funciones de grupo disponibles
- Describir el uso de las funciones de grupo
- Agrupar datos mediante la cláusula GROUP BY
- Incluir o excluir filas agrupadas utilizando la cláusula HAVING



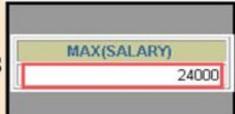
¿Qué son las Funciones de Grupo?

Las funciones de grupo operan en juegos de filas para dar un resultado por grupo.

EMPLOYEES

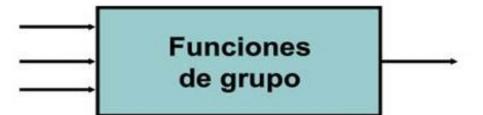
| DEPARTMENT_ID | SALARY |
|---------------|--------|
| 90 | 24000 |
| 90 | 17000 |
| 90 | 17000 |
| 60 | 9000 |
| 60 | 6000 |
| 60 | 4200 |
| 50 | 5800 |
| 50 | 3500 |
| 50 | 3100 |
| 50 | 2600 |
| 50 | 2500 |
| 80 | 10500 |
| 80 | 11000 |
| 80 | 8600 |
| | 7000 |
| 10 | 4400 |

Salario máximo en la tabla EMPLOYEES



Tipos de Funciones de Grupo

- AVG
- COUNT
- MAX
- MIN
- STDDEV
- SUM
- VARIANCE



Funciones de Grupo: Sintaxis

```
SELECT [column,] group function(column), ...

FROM table
[WHERE condition]
[GROUP BY column]
[ORDER BY column];
```

Uso de las Funciones AVG y SUM

Puede utilizar AVG y SUM para datos numéricos.

```
SELECT AVG(salary), MAX(salary),
MIN(salary), SUM(salary)

FROM employees
WHERE job_id LIKE '%REP%';
```

| AVG(SALARY) | MAX(SALARY) | MIN(SALARY) | SUM(SALARY) |
|-------------|-------------|-------------|-------------|
| 8150 | 11000 | 6000 | 32600 |

Uso de las Funciones MIN y MAX

Puede utilizar MIN y MAX para tipos de datos numéricos, de carácter y de fecha.

SELECT MIN(hire date), MAX(hire date)
FROM employees;

| MIN(HIRE_ | MAX(HIRE_ |
|-----------|-----------|
| 17-JUN-87 | 29-JAN-00 |

Uso de la Función COUNT

COUNT (*) devuelve el número de filas en una tabla:

SELECT COUNT(*)

FROM employees
WHERE department_id = 50;

COUNT(*)

COUNT (expr) devuelve el número de filas con valores no nulos para expr:

SELECT COUNT (commission_pct)
FROM employees
WHERE department_id = 80;

COUNT(COMMISSION_PCT)

3

Uso de la Palabra Clave DISTINCT

- COUNT (DISTINCT expr) devuelve el número de valores no nulos distintos de expr.
- Para mostrar el número de valores de departamento distintos en la tabla EMPLOYEES:

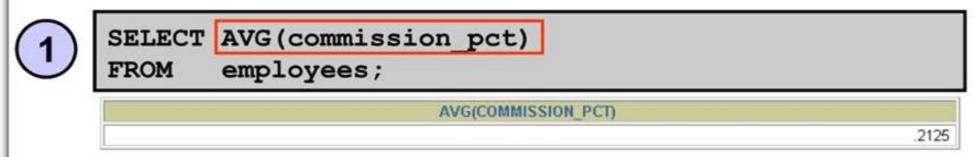
```
SELECT COUNT (DISTINCT department id)
FROM employees;
```

COUNT(DISTINCTDEPARTMENT_ID)

7

Funciones de Grupo y Valores Nulos

Las funciones de grupo ignoran los valores nulos en la columna:



La función NVL fuerza a las funciones de grupo a incluir valores nulos:



Creación de Grupos de Datos

EMPLOYEES

| | SALARY | DEPARTMENT_ID |
|-------|--------|---------------|
| 4400 | 4400 | 10 |
| | 13000 | 20 |
| 9500 | 6000 | 20 |
| | 5800 | 50 |
| | 3500 | 50 |
| 3500 | 3100 | 50 |
| | 2500 | 50 |
| | 2600 | 50 |
| | 9000 | 60 |
| 6400 | 6000 | 60 |
| | 4200 | 60 |
| | 10500 | 80 |
| 10033 | 8600 | 80 |
| 10000 | 11000 | 80 |
| | 24000 | 90 |
| | 17000 | 90 |

Salario medio de la tabla **EMPLOYEES** para cada departamento

| DEPARTMENT_ID | AVG(SALARY) |
|---------------|-------------|
| 10 | 4400 |
| 20 | 9500 |
| 50 | 3500 |
| 60 | 6400 |
| 80 | 10033.3333 |
| 90 | 19333.3333 |
| 110 | 10150 |
| | 7000 |

Creación de Grupos de Datos: Sintaxis de la Cláusula GROUP BY

```
SELECT column, group_function(column)

FROM table

[WHERE condition]

[GROUP BY group_by_expression]

[ORDER BY column];
```

Puede dividir las filas de una tabla en grupos más pequeños mediante la cláusula GROUP BY.

Uso de la Cláusula GROUP BY

Todas las columnas de la lista SELECT que no sean funciones de grupo se deben incluir en la cláusula GROUP BY.

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department id ;
```

| DEPARTMENT_ID | AVG(SALARY) |
|---------------|-------------|
| 10 | 4400 |
| 20 | 9500 |
| 50 | 3500 |
| 60 | 6400 |
| 80 | 10033.3333 |
| 90 | 19333.3333 |
| 110 | 10150 |
| | 7000 |

Uso de la Cláusula GROUP BY

La columna GROUP BY no tiene que estar en la lista SELECT.

```
SELECT AVG(salary)
FROM employees
GROUP BY department id ;
```

| AVG(SALARY) | |
|-------------|------------|
| | 4400 |
| | 9500 |
| | 3500 |
| | 6400 |
| | 10033.3333 |
| | 19333.3333 |
| | 10150 |
| | 7000 |

Agrupación por Más de Una Columna

EMPLOYEES

| DEPARTMENT_ID | JOB_ID | SALARY |
|---------------|------------|--------|
| 90 | AD_PRES | 24000 |
| 90 | AD_VP | 17000 |
| 90 | AD_VP | 17000 |
| 60 | IT_PROG | 9000 |
| 60 | IT_PROG | 6000 |
| 60 | IT_PROG | 4200 |
| 50 | ST_MAN | 5800 |
| 50 | ST_CLERK | 3500 |
| 50 | ST_CLERK | 3100 |
| 50 | ST_CLERK | 2600 |
| 50 | ST_CLERK | 2500 |
| 80 | SA_MAN | 10500 |
| 80 | SA_REP | 11000 |
| 80 | SA_REP | 8600 |
| | | |
| 20 | MK_REP | 6000 |
| 110 | AC_MGR | 12000 |
| 110 | AC_ACCOUNT | 8300 |

Sume los
salarios de
la tabla
EMPLOYEES para
cada puesto,
agrupados por
departamento

| DEPARTMENT_ID | JOB_ID | SUM(SALARY) |
|---------------|------------|-------------|
| 10 | AD_ASST | 4400 |
| 20 | MK_MAN | 13000 |
| 20 | MK_REP | 6000 |
| 50 | ST_CLERK | 11700 |
| 50 | ST_MAN | 5800 |
| 60 | IT_PROG | 19200 |
| 80 | SA_MAN | 10500 |
| 80 | SA_REP | 19600 |
| 90 | AD_PRES | 24000 |
| 90 | AD_VP | 34000 |
| 110 | AC_ACCOUNT | 8300 |
| 110 | AC_MGR | 12000 |
| | SA_REP | 7000 |

13 rows selected.

Uso de la Cláusula GROUP BY en Varias Columnas

```
SELECT department_id dept_id, job_id, SUM(salary)
FROM employees
GROUP BY department_id, job_id
```

| DEPT_ID | JOB_ID | SUM(SALARY) |
|---------|------------|-------------|
| 10 | AD_ASST | 4400 |
| 20 | MK_MAN | 13000 |
| 20 | MK_REP | 6000 |
| 50 | ST_CLERK | 11700 |
| 50 | ST_MAN | 5800 |
| 60 | IT_PROG | 19200 |
| 80 | SA_MAN | 10500 |
| 80 | SA_REP | 19600 |
| 90 | AD_PRES | 24000 |
| 90 | AD_VP | 34000 |
| 110 | AC_ACCOUNT | 8300 |
| 110 | AC_MGR | 12000 |
| | SA_REP | 7000 |



Consultas llegales que Utilizan Funciones de Grupo

Cualquier columna o expresión de la lista SELECT que no sea una función de agregación se debe incluir en la cláusula GROUP BY:

```
SELECT department_id, COUNT(last_name)
FROM employees;
```

```
SELECT department_id, COUNT(last_name)

*
ERROR at line 1:
ORA-00937: not a single-group group function
```

Falta una columna en la cláusula GROUP BY



Consultas llegales que Utilizan Funciones de Grupo

- No puede utilizar la cláusula WHERE para restringir grupos.
- Utilice la cláusula HAVING para restringir grupos.
- No puede utilizar funciones de grupo en la cláusula WHERE.

```
SELECT department_id, AVG(salary)
FROM employees
WHERE AVG(salary) > 8000
GROUP BY department_id;
```

```
WHERE AVG(salary) > 8000

*
ERROR at line 3:
ORA-00934: group function is not allowed here
```

No se puede utilizar la cláusula WHERE para restringir grupos

Restricción de Resultados de Grupos

EMPLOYEES

| DEPARTMENT_ID | SALARY | | |
|---------------|--------|--|--|
| 90 | 24000 | | |
| 90 | 17000 | | |
| 90 | 17000 | | |
| 60 | 9000 | | |
| 60 | 6000 | | |
| 60 | 4200 | | |
| 50 | 5800 | | |
| 50 | 3500 | | |
| 50 | 3100 | | |
| 50 | 2600 | | |
| 50 | 2500 | | |
| 80 | 10500 | | |
| 80 | 11000 | | |
| 80 | 8600 | | |
| | | | |
| 20 | 6000 | | |
| 110 | 12000 | | |
| 110 | 8300 | | |

Salario máximo por departamento cuando es superior a 10.000 dólares

| DEPARTMENT_ID | MAX(SALARY) |
|---------------|-------------|
| 20 | 13000 |
| 80 | 11000 |
| 90 | 24000 |
| 110 | 12000 |

Restricción de Resultados de Grupos con la Cláusula HAVING

Si se utiliza la cláusula HAVING, Oracle Server restringe los grupos así:

- Se agrupan las filas.
- 2. Se aplica la función de grupo.
- Se muestran los grupos que satisfacen la cláusula HAVING.

```
SELECT column, group_function

FROM table

[WHERE condition]

[GROUP BY group_by_expression]

[HAVING group_condition]

[ORDER BY column];
```

Uso de la Cláusula HAVING

```
SELECT department_id, MAX(salary)
FROM employees
GROUP BY department id
HAVING MAX(salary)>10000 ;
```

| DEPARTMENT_ID | MAX(SALARY) |
|---------------|-------------|
| 20 | 13000 |
| 80 | 11000 |
| 90 | 24000 |
| 110 | 12000 |

Uso de la Cláusula HAVING

```
SELECT job_id, SUM(salary) PAYROLL
FROM employees
WHERE job_id NOT LIKE '%REP%'
GROUP BY job id
HAVING SUM(salary) > 13000
ORDER BY SUM(salary);
```

| JOB_ID | PAYROLL |
|---------|---------|
| IT_PROG | 19200 |
| AD_PRES | 24000 |
| AD_VP | 34000 |

Anidamiento de Funciones de Grupo

Muestre al salario medio máximo:

```
SELECT MAX(AVG(salary))
FROM employees
GROUP BY department_id ;
```

MAX(AVG(SALARY))

19333.3333



Resumen

En esta lección ha aprendido a:

- Utilizar las funciones de grupo COUNT, MAX, MIN y AVG
- Escribir consultas que utilicen la cláusula GROUP BY
- Escribir consultas que utilicen la cláusula HAVING

```
SELECT column, group_function

FROM table

[WHERE condition]

[GROUP BY group_by_expression]

[HAVING group condition]

[ORDER BY column];
```

Práctica 4: Visión General

Esta práctica cubre los temas siguientes:

- Escritura de consultas que utilicen las funciones de grupo
- Agrupación por filas para obtener más de un resultado
- Restricción de grupos mediante la cláusula HAVING

